CmC SOFTWARE

CmC SOFTWARE

Word Processor Program

CONNECTICUT microCOMPUTER
BROOKFIELD, CONNECTICUT 06804

TABLE OF CONTENTS

I. Important 1

II. Introduction 2

III. Word Processor Terms and Their Definitions 3

IV. Getting Acquainted with the CmC Word Processor 5

V. Summary of Commands 10

VI. Summary of Directives 11

VII. Reference manual - Commands 12

VIII. Reference manual - Directives 19

IX. Errors and Crashes 21

X. Examples 22

I. IMPORTANT

The CONNECTICUT microCOMPUTER WORD PROCESSOR PROGRAM and this manual are copyrighted by CONNECTICUT microCOMPUTER and may not be copied, in whole or in part, without the written permission of CONNECTICUT microCOMPUTER.

****

The CmC WPP is guaranteed to be loadable into your PET. If, after trying several times, you cannot successfully load the WPP, return your cassette with a description of the problem and we will send you another cassette. If the new tape will not load we will refund your purchase price.

<><><><><><><><><><><><><><><><><><><><><><><><><><><><><><>

If, at any time while running the WPP, you inadvertantly enter the PET BASIC mode, type

    GOTO 1000 (RETURN)

to return to the WPP without losing your text.

VVV\/\/\/VVV\/\/\/VVV\/\/\/VVV\/\/\/VVV\/\/\/VVV\/\/\/

Special function keys:

The DEL key deletes the previous character.
The HOME key cancels the current line and returns you to the command and edit mode.
The CLR key (shift HOME) cancels the current line, returns you to the command and edit mode, and clears the screen.

In the input mode, for the printer:

The left cursor is the backspace key.
The left arrow is the underline key.

II. INTRODUCTION

Word processing refers to the generation, editing, and printing of text files. Word processing machines vary in complexity from simple duplicating typewriters with paper tape punches and readers, to the large mexicomputers with megabytes of disk storage. The simplest machines allow single pages of text to be put on paper tape for duplicating without retyping. Directives may be nonexistent or may allow simple functions such as specifying line length or turning the typewriter off. Editing is limited to cutting and splicing paper tape sections.

The most complex machines allow documents hundreds of pages long to be saved and generated. Editing features include changing words and phrases, searching, moving sections, and merging files. Directives allow justification, centering, skipping, paging, and so on.

The CmC Word Processor falls somewhere between these two extremes. The CmC WPP was written to be used with an 8K Commodore PET, a CmC printer adapter, and an RS-232 printer. The storage medium for the files is magnetic cassette tape. Files may be saved and merged. Lines may be inserted, deleted, and moved. Directives set left margins, line length, centering, etc.

The program, itself, occupies approximately 4890 bytes and needs approximately 300 bytes for buffers. This leaves almost 2000 bytes for user text. This is enough for about a one page document.

The WPP, as delivered, allows 100 lines of text. If your PET has additional memory, change line 200 of the WPP to read

```
200 Y=250.
```

This will allow 250 lines of text.

III. WORD PROCESSOR TERMS AND THEIR DEFINITIONS

This chapter presents the definitions of terms as they relate to the CmC Word Processor Program.

break — An action that causes the contents of the output buffer to be printed and the output buffer to be cleared.

command — A statement entered from the keyboard to tell the WPP what function to perform.

current line — The textline being pointed to by the current line pointer. To find out which line is being pointed to type "t" and press the RETURN key.

directive — A command in the textfile used by the WPP, when printing a document, to format the document. Directives make a word processor more than just a text editor. The directives are commands that are used by the word processor to format the rawtext as a document is printed. Directives specify document line length, margins, centering, line spacing, etc. Directives also permit underlining, transmission of control characters, etc. The textfile consists of rawtext and directives. The rawtext is printed on the document in a format specified by the directives. A directive is identified by the occurrence of a blank or period at the beginning of a textline.

display — Put on the PET screen.

document — The finished page, letter, etc. produced by the print command of the WPP.

down — Refers to the PET screen - increasing line numbers.

mode — A manner of operation of the WPP. There are three modes of operation for the WPP. These are the input mode, command and edit mode, and print mode. In the input mode, as lines are typed in from the keyboard, they are put into the textfile. In the command and edit mode, as lines are typed in from the keyboard, the WPP interprets each line and performs a specific function (command). In the print mode, the WPP reads the textfile in the PET memory, formats the text, and prints a document.

output buffer — The temporary storage area where a line is composed before being sent to the printer or screen.
When the Word Processor Program is commanded to print, it constructs a complete line of rawtext, storing it in the output buffer, before it starts printing. The size of the output buffer is equal to the line length. As a line of rawtext of the textfile is read during execution of the print command, each word is appended to the output buffer, up to the line length. No parts of words are added, so that the output buffer may not be filled before printing. If a word would overflow the buffer, then the buffer is printed, cleared, and the new word is then put into the beginning of the buffer.
The output buffer is also printed and cleared each time a directive or a space at the beginning of a text line is encountered.

print — Refers to the document (processed textfile) format.

rawtext — The printed portion of the textfile. The textfile consists of rawtext and directives. The rawtext is printed on the document in a format specified by the directives.

textfile — The file of rawtext and directives written in the input mode and used by the WPP to generate the document.

textline — One line of the textfile.

type — Refers to the textfile format.

up — Refers to the PET screen - decreasing line numbers.

IV. GETTING ACQUAINTED WITH THE CmC WORD PROCESSOR PROGRAM (WPP)

This chapter is intended to help you become familiar with the CmC Word Processor Program (WPP). Perform each of the operations as they are presented. Work slowly and carefully. Study these operations and the results they produce. Review the previous chapter on WPP terms and refer to it as you go through this chapter.

The symbol ↵ means "press the RETURN key".
The symbol ␣ means "press the SPACE key".
The DEL key deletes the previous character.
The HOME key cancels the current line and returns you to the command and edit mode.
The CLR key (shift HOME) cancels the current line, returns you to the command and edit mode, and clears the screen.

In the Input mode, for the printer:
The left cursor is the backspace key.
The left arrow is the underline key.

Load the WPP into the PET and type "RUN↵" (type the letters R, U, and N followed by pressing the RETURN key - don't type the quotes). The screen will clear and a prompt character (>) will appear on the top left of the screen. You are now in the command and edit mode of the WPP. This means that you can load textfiles from the cassette unit, edit the textfile, print a document, type the textfile, save the textfile on cassette, and enter the input mode.

Since typing RUN clears the BASIC variables no textfile currently exists in the PET memory. To input a textfile from the PET keyboard you must enter the WPP's INPUT mode. To do this type "input↵". On the screen will appear

```
Input mode.
 Ø>
```

The "Ø" is the current line number. The ">" is the prompt character. Anything typed in now, before the RETURN key is pressed, will become part of line Ø of the textfile.

Now, type

```
 Ø>This is line zero.↵
```

(Type only the underlined portion and don't forget the spaces.)

On the screen will appear

```
 1>
```

Now, following each line number and prompt character, type in the lines

```
1>This is line one.↵
2>This is line two.↵
3>This is line three.↵
4>This is line four.↵
5>This is line five.↵
6>This is line six.↵
7>↵
```

If the RETURN is the only key pressed on an INPUT line, you will leave the INPUT mode and return to the command and and edit mode. A ">" without a number should now be on the left side of the screen. The WPP is now ready for your next command.

Now type "type↵". You should see

```
6>This is line six.
```

This is the last line you entered into the textfile. Now, type "goto␣2↵". On the screen you should see

```
2>This is line two.
```

The "goto␣N" command sets the current line pointer to N and then types the current line.

Type "goto↵". This is the same as typing "goto␣0↵". You should see

```
0>This is line zero.
```

Now, type "type␣3↵".

The first three textlines will be typed on the screen. Now, type "type␣3↵". Text lines 2, 3 and 4 will be typed. The current line pointer is set to the last line typed. Now, type "type␣15↵". Only textlines 4 through 6 will be typed since line 6 is the last line of the textfile.

Type "g␣2↵". ("goto" is abbreviated "g". Don't forget the space between the "g" and "2".) Type "i↵". ("input" is abbreviated "i".) On the screen will appear

```
Input mode.
 3>
```

You have now reentered the input mode. The next lines you input will become part of the textfile and will be inserted between lines 2 and 3 of the original file. Type in the lines

```
3>.ll 20↵      (Type only the underlined portion and
4>.ls 3↵              don't forget the spaces.)
5>↵
```


You should now be back in the command and edit mode. Type "t↵". You should see

```
 4>.ls 3

This is the last line you entered.
```

Type "down↵" (or "d↵"). You should see

```
 5>This is line three.
```

Now, type "g↵" and "t␣25↵".

The entire textfile should now be typed on the PET screen. Line 3 sets the output document line length to a maximum of 20. (The line length is 60 if you don't specify another number.) Line 4 sets the line spacing to 3. (The line spacing is 1 if you don't specify another number.)

Now, type "g↵", "print↵". The textfile will now be printed instead of typed. This means the line numbers will not appear, the textlines will be combined to produce document lines up to the line length, and all directives will be followed. The current line pointer is not changed.

To get hard copy, connect up your printer to the PET using your CmC ADApter. Now, type "print␣10␣5↵". This means: print 10 textlines, starting with the current line, on output device number 5. The document printed should look like

```
This is line zero. This is line one. This is line two.
This is line three.


This is line four.


This is line five.


This is line six.
```

After the document is printed type "print␣␣5". This means: print all the lines of the textfile, starting with the current line, on device number 5. The document printed should look like the previous document.

Now, type "g␣4↵". You should see.

```
 4>.ls 3
```

Type "change␣/ls/sk/↵". You should now see

```
4>.sk 3
```

This command changed the string "ls" of the textline number 4, to the string "sk". The delimiter between strings, in this case, is "/". Any character can be used as the delimiter.

Now, type "pØ5R". Printing starts with textline 4.

Now, type "gR", "pØ5R". Compare the documents just printed. Observe that the ".sk" directive causes line skipping only once while the ".ls" directive causes line skipping (spacing) until another ".ls" directive is read.

To type the textfile on the printer, type "tØ2Ø5". If an equal sign is printed after each line number, ignore it. This is how the printer interprets the cursor left in the command and edit mode. The printout should look like

```
Ø=>This is line zero.
1=>This is line one.
2=>This is line two.
3=>.ll 20
4=>.sk 3
5=>This is line three.
6=>This is line four.
7=>This is line five.
8=>This is line six.
```

Put a blank cassette into the PET. Type "saveR" and then press the PLAY and REC buttons on tape recorder #1. The WPP will save all nine lines of the textfile on the cassette.

When the cursor returns, press the STOP and then the REWIND keys on the cassette recorder. When the tape is rewound press the STOP key on the recorder again.

Type "gØ3R. The current line pointer is now at line 3. Now, type "eraseØ2R". Lines 2 and 3 will be erased and the rest of the file will be moved up to fill in the gap. Look at the entire textfile by typing "gR" and "tØ1ØR".

The current line pointer is now at line 6, the last line of the textfile. Move the current line pointer to line 1 by typing "upØ5R". Type "loadR" and press the PLAY key on the tape recorder. As the textfile is inputted from the tape, note that the WPP starts inputting at line 2.

When the cursor returns type "tR". The last line inputted from the cassette will be displayed. Type "downR". The next line will be displayed. This is the line that was line 2 before we inputted from the cassette.

Type "gR" and "tØ2ØR" to display the entire textfile. Observe how the textfile was loaded from the tape and inserted into the old textfile without losing any of the old textfile.

If you have followed the directions in this section, then you have created a textfile, performed some editing functions, typed, printed and saved and loaded textfiles on cassette. There are more features of the CmC Word Processor Program. There are commands and directives that we have not covered in this section. Read and study the rest of this manual. All of the commands and directives are summarized and explained in detail.

V. SUMMARY OF COMMANDS

change&/S1/S2/	Change the string S1 of the current line to the string S2
down&N	Move the current line pointer down N lines
erase&N	Erase N lines starting with the current line
free	Same as the PET "PRINT FRE(X)" command
goto&N	Set the current line pointer to line N
input	Enter the input mode.
load	Load the next textfile on the cassette tape
move&N1&N2	Move the block of text from line N1 through line N2 to after the current line
print&N1&N2&N3	Print N1 lines, starting with the current line, onto device N2 and repeat N3 times
replace	Replace the current line with the next typed line
save	Save the textfile on the PET cassette
type&N1&N2	Type N1 lines, starting with the current line onto device N2
up&N	Move the current line pointer up N lines
x	Returns the line length of the current line

VI. SUMMARY OF DIRECTIVES

All directives must start in the first column of a line or it will be treated as text.
All directives cause a break: The output buffer is printed and cleared.

.ac&N1&N2	Send the character corresponding to the ASCII code number.
.ce&S	Center and print the string S.
.ll&N	Set the line length to N.
.lm&N	Set the left margin to N.
.ls&N	Set the line spacing to N.
.nl	The next time the output buffer is printed, no line feed is issued.
.pa	Pause the WPP until any key is pressed.
.rc&N	Repeat the string on the next textline N times, then center and print the string.
.sk&N	Skip N lines.
.	Break. Print and clear the output buffer.
&S	Print and clear the output buffer, then put a space and the string S into the beginning of the buffer.

VII. Reference Manual - Commands

Commands are issued in the command and edit mode. Commands tell the Word Processor Program (WPP) what action to perform. One command is issued per line.

In this chapter, each command is presented along with its abbreviation and required format. The WPP does not ignore spaces (ƀ) in commands.

<><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><>
<><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><>

Command: changeƀ/S1/S2/
cƀ/S1/S2/
cƀ/S1/S2

Action: Changes the string S1 of the current line to the string S2. The slash (/) is the string delimiter and any other character may also be used. The last delimiter may be left off. If S2 is not specified, the string S1 is erased.

Example 1:
```
>t
4>This line is rwong
>c /rwong/right.
4>This line is right.
>c /line /
4>This is right.
```
Example 2:
```
>t
9>seen by
>c //not
9>not seen by
```

<><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><>
<><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><>

Command: downƀN
dƀN
d

Action: Moves the current line pointer down N lines. If N is not specified, the current line pointer is moved down one line.

Example :
```
>t
5>.lm 30
>down
6>.ce FINALS
>d 3
9>first place is
```


<><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><>
<><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><>

Command: eraseƀN
erƀN
er

Action: Erase N lines starting with the current line. If N is not specified only the current line is erased. The remaining lines are moved up to replace the erased lines.

Example :
```
>t 1Ø
Ø>zero
1>one
2>two
3>three
4>four
5>five
6>six
>g 2
2>two
>er 2
2>four
>g
Ø>zero
>t 1Ø
Ø>zero
1>one
2>four
3>five
4>six
```

<><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><>
<><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><>

Command: free
f

Action: Prints, onto the screen, the number of characters of available memory. This is the same function as the PET command PRINT FRE(X).

Each character of the textfile uses up one character of the available memory space.

Some of the available memory will be used by the Word Processor Program's buffers so that the textfile must be something less than the available memory space. How much less depends on which commands are used, line lengths, etc.

Example :
```
>f
1683
```

<><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><>
<><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><>

```
Command: gotoɃN
         gɃN
         g
```

Action: Sets the current line pointer to N and types the current line. If N is not specified the current line pointer is set to Ø (the top of the textfile).

```
Example  : >t
            5>a line of
           >goto 17
            17>Sold for $324.95 on the open
           >g
            Ø>.11 72
```

<><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><>
<><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><>

```
Command: input
         i
```

Action: Enters the input mode in preparation for inserting lines into the textfile.

When the input mode is entered a number and prompt character (>) followed by the cursor appear on the PET screen. Anything now typed before the next Ɍ will go into the textfile. The numbers to the left of the prompt character are the textline numbers.

To exit the input mode, type a Ɍ without anything else in the textline. The current line pointer will now point to the last textline entered into the textfile.

Inputting starts with the textline number after the current line. If the textfile is empty, the inputting starts at line Ø. If lines are inserted in the middle of a textfile, the old textlines after the old current line are moved down.

To input at the beginning of the textfile (before the old textline Ø) type "gɌ", "iɌ".

To input after the old textline Ø, type "gɌ","tɌ","iɌ".

<><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><>
<><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><>

```
Command: load
         l
```

Action: Loads the next textfile from the cassette into the textfile in the PET memory. The cassette textfile is inserted into the old textfile after the current line pointer.

<><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><>
<><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><>

```
Command: moveɃN1ɃN2
         mɃN1ɃN2
         mɃN1
```

Action: Moves the block of lines starting at line number N1 through line number N2, to after the current line. If N2 is not specified, or is less than or equal to N1, then one line is moved. If N1 or N2 is greater than the line number of the last line of the text file, then the error message

??!!

is displayed and no move takes place.

```
Example  : >t 15
            Ø>line Ø
            1>line 1
            2>line 2
            3>line 3
            4>line 4
            5>line 5
           >g 1
            1>line 1
           >m 3 4
           >g
            Ø>line Ø
            1>line 1
            2>line 3
            3>line 4
            4>line 2
            5>line 5
```

<><><><><><><><><><><><><><><><><><><><><><><><><><><><><><>
<><><><><><><><><><><><><><><><><><><><><><><><><><><><><><>

```
Command: printƀN1ƀN2ƀN3
         pƀN1ƀN2ƀN3
         pƀƀN2ƀN3
         pƀƀƀN3
         pƀN1ƀN2
         pƀN1
         p
```

Action: Prints N1 lines of the textfile, starting at the current line, onto device N2, and prints the document N3 times. If N1 is omitted, the rest of the textfile, starting with the current line, is printed.

If N2 is omitted, or is 3, the printing is done on the PET screen. If N3 is omitted, the document is printed once.

If printing starts with a line other than the first line of the textfile, all directives (e.g. line length, left margin) are read and followed starting with textfile line number Ø.

Example 1: >p 15 5 3

Prints 15 lines of the textfile, starting with the current line, on device number 5, three times.

Example 2: >p

Prints the textfile from the current line to the last line onto the PET screen.

<><><><><><><><><><><><><><><><><><><><><><><><><><><><><><>
<><><><><><><><><><><><><><><><><><><><><><><><><><><><><><>

```
Command: replace
         r
```

Action: The current line is replaced by the line typed following the command.

```
Example :  >t
            66>.rc STRING
           >r
            66>.ac 4
            66>.ac 4
           >t
            66>.ac 4
```


<><><><><><><><><><><><><><><><><><><><><><><><><><><><><><>
<><><><><><><><><><><><><><><><><><><><><><><><><><><><><><>

```
Command: save
         s
```

Action: The textfile in the PET memory is saved on the tape cassette in the PET cassette recorder.

```
Example  : >s
           PRESS PLAY & RECORD ON TAPE #1
           OK
           >
```

<><><><><><><><><><><><><><><><><><><><><><><><><><><><><><>
<><><><><><><><><><><><><><><><><><><><><><><><><><><><><><>

```
Command: typeƀN1ƀN2
         tƀN1ƀN2
         tƀƀN2
         t
```

Action: Type N1 lines of the textfile, starting at the current line, onto device N2. If N1 is omitted one line is typed. If N2 is omitted the outpyut device is the PET screen (device number 3). The current line pointer is set to the number of the last line typed.

```
Example 1: >t
            5>a line
           >t
            5>a line
           >t 3
            5>a line
            6>another line
            7>a third line
```

Example 2: >t 100 5

The entire textfile, starting with the current line, is typed on device number 5.

<><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><>
<><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><>

Command: upßN
ußN
u

Action: Moves the current line pointer up N lines. If N is omitted, the current line pointer moves up one line.

```
Example  : >t
            11>but not second.
           >up 2
            9>First place is
           >u
            8>.ce FINALS
           >u 3
            5>.lm 6
```

<><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><>
<><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><><>

Command: x

Action: Types the length of the current line onto the PET screen.

This function is helpful in determining if there are spaces at the end of a line, or if there are control characters, such as backspaces, in a line.

VIII. Reference Manual - Directives

Directives are part of the textfile. The actions they specify are performed only when the print command of the Word Processor Program (WPP) is issued. A directive must start in the first column of a line and only one directive is allowed per line.

In this chapter each directive is presented along with its required format. The directives may not be abbreviated. The WPP does not ignore spaces in directives.

=#==#==#==#==#==#==#==#==#==#==#==#==#==#==#==#==#=

Directive: .acßN1ßN2

Action: ASCII character. Causes a break and sends the character corresponding to the ASCII code N1, N2 times.

Example 1> .acß7ß3
Rings the bell on the printer (if it has one), three times.

Example 2> .acß4
Sends an EOT character (End Of Transmission) which turns off some printers.

=#==#==#==#==#==#==#==#==#==#==#==#==#==#==#==#==#=

Directive: .ceßS1

Action: Center. Causes a break then centers and prints the string S1.

=#==#==#==#==#==#==#==#==#==#==#==#==#==#==#==#==#=

Directive: .llßN

Action: Line length. Causes a break and then sets the output line length to N.

=#==#==#==#==#==#==#==#==#==#==#==#==#==#==#==#==#=

Directive: .lmßN

Action: Left margin. Causes a break and then sets the left margin to N. The line length does not include the left margin.

=#==#==#==#==#==#==#==#==#==#==#==#==#==#==#==#==#==#=

Directive: .ls⌀N

Action: Line spacing. Causes a break and then sets the line spacing to N.

=#==#==#==#==#==#==#==#==#==#==#==#==#==#==#==#==#==#=

Directive: .nl

Action: No line feed. Causes a break and suppresses the line feed the next time the output buffer is printed. This directive is useful for underlining and overlaying.

=#==#==#==#==#==#==#==#==#==#==#==#==#==#==#==#==#==#=

Directive: .pa

Action: Pause. Causes a break and then a pause. The Word Processor Program halts all operations until a key is depressed on the keyboard. If your printer has a keyboard, the Word Processor Program can print to some point of the text and stop. Printing can then be done from the keyboard and then resumed by the Word Processor Program.

=#==#==#==#==#==#==#==#==#==#==#==#==#==#==#==#==#==#=

Directive: .rc⌀N

Action: Repeat and center. Causes a break and then reads the next line of the textfile N times into the output buffer. The output buffer is centered and printed. If N is not specified the entire output buffer is filled.

=#==#==#==#==#==#==#==#==#==#==#==#==#==#==#==#==#==#=

Directive: .sk⌀N

Action: Skip. Prints and clears the output buffer and then skips N lines. If N is 0 or not specified, then one line is skipped.

=#==#==#==#==#==#==#==#==#==#==#==#==#==#==#==#==#==#=

Directive: ⌀S1

Action: Break. Prints and clears the output buffer and then puts a space and the string S1 into the output buffer.

=#==#==#==#==#==#==#==#==#==#==#==#==#==#==#==#==#==#=

Directive: .

Action: Break. Prints and clears the output buffer.

IX. ERRORS AND CRASHES

The CmC Word Processor Program was designed so that once the program is running, the user never returns to the PET Basic environment. Any errors by the user should produce an error message rather than a crash.

The ideal is rarely the case. Some conditions, such as an out of memory error, will return the user to the PET Basic. In this case type

GOTO 1000↵.

This will return to the WORD PROCESSOR without losing the textfile.

Typing

RUN↵

will return the user to the Word Processor but will clear all variables including the textfile.

Pressing the RUN-STOP key will cause the the PET to leave the Word Processor Program and to enter the PET Basic environment. Typing

GOTO 1000↵

will return to the Word Processor without destroying the textfile.

X. EXAMPLES

()-

Example 1:

This manual was compiled on the COMMODORE PET and was printed on a GE TermiNet 300 using a CmC ADA 1200C printer adapter and the CmC Word Processor Program. The textfile for the Table of Contents and the first chapter is printed below.

```
0=>.sk 3
1=>.ll 65
2=>.ce TABLE OF CONTENTS
3=>.rc 17
4=>-
5=>.ls 3
6=>.sk
7=>    I. Important
8=>   II. Introduction
9=>  III. Word Processor Terms and Their Definitions
10=>  IV. Getting Acquainted with the CmC Word Processor
11=>   V. Summary of Commands
12=>  VI. Summary of Directives
13=> VII. Reference manual - Commands
14=>.
15=>VIII. Reference manual - Directives
16=>  IX. Errors and Crashes
17=>   X. Examples
18=>.ls
19=>.sk 5
20=>.ce I. IMPORTANT
21=>.rc 12
22=>-
23=>.sk
24=>      The CONNECTICUT microCOMPUTER WORD PROCESSOR PROGRAM
25=>and this manual
26=>are copyrighted by CONNECTICUT microCOMPUTER
27=>and may not be copied, in whole or in part, without
28=>the written permission of CONNECTICUT microCOMPUTER.
29=>.sk
30=>.rc
31=>*+
32=>.sk
33=>      The CmC WPP is guaranteed to be loadable into your PET.
34=>If, after trying several times, you cannot successfully load the WPP,
35=>return your cassette with a description of the problem
36=>and we will send you another cassette.
37=>If the new tape will not load we will refund your purchase price.
38=>.sk
39=>.rc
40=><>
41=>.sk
42=>      If, at any time while running the WPP,
43=>you inadvertantly enter the PET BASIC mode, type
44=>         GOTO 1000 (RETURN)
45=>.
46=>to return to the WPP without losing your text.
47=>.sk
48=>.rc
49=>\/
50=>.sk
51=>      Special function keys:
52=>.sk
53=>        The DEL key deletes the previous character.
54=>        The HOME key cancels the current line and returns
55=>           you to the command and edit mode.
56=>        The CLR key (shift HOME) cancels the current line,
57=>           returns you to the command and edit mode, and
58=>           clears the screen.
59=>.sk
60=>      In the input mode, for the printer:
61=>.sk
62=>        The left cursor is the backspace key.
63=>        The left arrow is the underline key.
64=>.sk 6
65=>.ac 4
```


()-()

Example 2:

A letter and its textfile are reproduced on the next two pages. All the directives are used. Note the pause directive at the beginning of the textfile that allows a salutation to be printed, and the ASCII character directive at the end of the textfile that turns off the GE Terminet 300 terminal.

CMC

CONNECTICUT microCOMPUTER

150 POCONO ROAD - BROOKFIELD, CONNECTICUT 06804

(203) 775-9659

August 7, 1978

Mr. Daniel Fylstra
Personal Software
P. O. Box
Cambridge, Mass. 02138

Dear Dan,

Thank you for your order. for our Word Processor Program. This letter was written using this software. The textfile that generates this letter is reproduced on the next page.

<><><><><><><><><><><>

All the currently
available directives are
used.

Overlays are possible.

The Word Processor Program is believed to be free of software bugs at this time. If you find any bugs or errors, please write us describing what you did, what the WPP did, and what you expected.

We are planning a more powerful version of the WPP to be used on a 16K PET with dual cassettes or floopy disk. If you have any comments or suggestions for additional commands, directives or other features, please write to us and let us know.

Are there other PET compatible products you would like to see? Let us know about these, too.

Be careful about spaces
at the beginning and at the ends of lines in the textfile.
A space at the beginning
of a line causes a break.
A space at the end of a line (as in textline 49) may cause two spaces between words in the document.

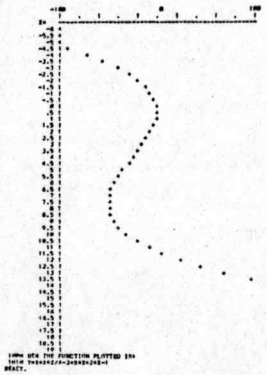
Sincerely,

Richard Rosner,
Vice President

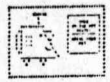
```
0=>.ll 70
1=>.lm 40
2=>August 7, 1978
3=>.sk
4=>.lm 5
5=>.pa
6=>.sk
7=>     Thank you for your order.
8=>for our Word Processor Program.
9=>This letter was written using this software.
10=>The textfile that generates this letter
11=>is reproduced on the next page.
12=>.sk 2
13=>.rc 20
14=><>
15=>.sk 2
16=>.lm 18
17=>.ll 25
18=>     All the currently available directives are used.
19=>.lm 5
20=>.ll 70
21=>.sk
22=>.nl
23=>.ce Overlays are possible.
24=>.ce _______ //_  //   _
25=>.sk
26=>     The Word Processor Program is believed to be free of software bugs
27=>this time. If you find any bugs or errors, please write us describing
28=>what you did, what the WPP did, and what you expected.
29=>.sk
30=>.ls 2
31=>     We are planning a more powerful version of the WPP
32=>to be used on a 16K PET with dual cassettes or floopy disk.
33=>If you have any comments or suggestions for additional commands,
34=>directives or other features, please write to us and let us know.
35=>.ls
36=>     Are there other PET compatible products
37=>you would like to see?
38=>Let us know about these, too.
39=>.sk
40=>Be careful
41=>about spaces
42=>.
43=>at the beginning
44=>and at the ends of lines
45=>in the textfile.
46=>.
47=>A space at the beginning
48=> of a line causes a break.
49=> A space at the end of a line
50=>(as in textline 49) may cause
51=>two spaces between words in the document.
52=>.sk
53=>.lm 40
54=>Sincerely,
55=>.sk 3
56=>Richard Rosner,
57=>   Vice President
58=>.ac 4
```

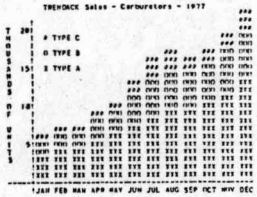
PET prints

COMMODORE PET HARD COPY OUTPUT USING PET ADA 1200

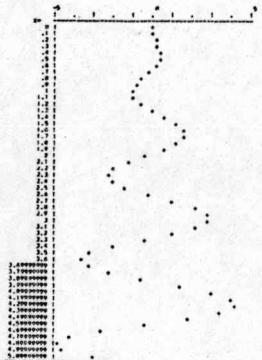


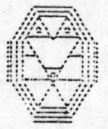
```
 1 OPEN 6,8:CMD 6:LIST
10 REM ARCSIN AND ARCCOS FUNCTIONS FOR THE COMMODORE PET
20 REM WRITTEN BY RICHARD WOTHERS
30 REM          CONNECTICUT MICROCOMPUTER
40 REM          150 POCONO ROAD
50 REM          BROOKFIELD, CT 06804
60 REM
70 REM *Listed on a GE TermiNet 300
80 REM *using a CmC ADA 1200.
90 REM
100 REM OPEN OUTPUT FILE ON DEVICE #8.
110 OPEN 5,8
120 REM
500 REM GET A NUM VALUE
510 INPUT S
520 C=S
530 REM
1000 REM THE SINE OF THE ANGLE IS S
1010 REM IF THE SIN IS IN THE RANGE OF -1 TO 1, THEN COMPUTE.
1020 IF S>1 AND S<-1 GOTO 1050
1030 A5=90*ATN(S/SQR(-S*S+1))
1040 REM THE ARCSINE IS AT
1050 A5=A5*ATN(S/SQR(-S*S+1))
1060 REM THE RESULT IS IN RADIANS. CONVERT TO DEGREES.
1070 A5=A5*180/
2000 REM THE COSINE OF THE ANGLE IS C
2010 REM IF THE COSINE IS IN THE RANGE OF -1 TO 1,
2011 REM      AND NOT = TO 0, THEN COMPUTE
2020 IF C>0 THEN 2040
2030 AC=90*ATN(1)*2000
2040 IF C<1 AND C<>1 THEN 2070
2050 AC=90*ATN(1)*2000
2060 REM THE ARCCOS IS AC
2070 AC=ATN(1-C*C)^.5/C)
2080 REM THE RESULT IS IN RADIANS. CONVERT TO DEGREES.
2090 AC=AC*180/
3000 PRINT#5," SIN,COS       ARCSIN        ARCCOS"
3010 PRINT#5,.5S
3020 FOR N=1 TO 13-LEN(STR$(S)):PRINT#5," ";:NEXT
3030 PRINT#5,A5S
3040 FOR N=1 TO 13-LEN(STR$(A5)):PRINT#5," ";:NEXT
3050 PRINT#5,AC
3100 GOTO 500
READY.
```

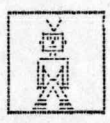


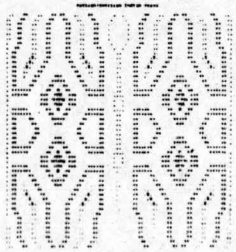


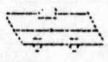
```
SIN,COS          ARCSIN          ARCCOS
.1               5.73917448      84.2608299
SIN,COS          ARCSIN          ARCCOS
.2               11.536959       78.463041
SIN,COS          ARCSIN          ARCCOS
.3               17.4578631      72.5421369
SIN,COS          ARCSIN          ARCCOS
.33333333        19.4712204      70.5287796
SIN,COS          ARCSIN          ARCCOS
.4               23.5781785      66.4218214
SIN,COS          ARCSIN          ARCCOS
.5               30              60
SIN,COS          ARCSIN          ARCCOS
.6               36.8698977      53.1301024
SIN,COS          ARCSIN          ARCCOS
.7               44.427096       45.572904
SIN,COS          ARCSIN          ARCCOS
.725811882       46.5446825      43.4553965
SIN,COS          ARCSIN          ARCCOS
.8               53.1301024      36.8698977
SIN,COS          ARCSIN          ARCCOS
.9               64.1580673      25.8419328
SIN,COS          ARCSIN          ARCCOS
.999999          89.9189618      .081382363
```











Powers of X

X	X^2	X^3	X^.25	X^.5
0	0	0	0	0
50	2500	125000	2.17586e+08	8.30083e+08	7.07107e+08
100	10000	1e+06	1.01948e+09	3.16228e+09	10
150	22500	3.375e+06	4.89391e+09	7.69753e+09	12.2474e+08
200	40000	8e+06	1.60391e+10	1.49336e+10	14.1421e+08
250	62500	1.5625e+07	3.41406e+10	2.80315e+10	15.8114e+08
300	90000	2.7e+07	6.09549e+10	5.19121e+10	17.3205e+08
350	1.225e+08	4.2875e+08	9.82718e+10	8.23173e+10	18.7083e+08
400	1.6e+08	6.4e+08	1.46484e+11	1.2574e+11	20
450	2.025e+08	9.1125e+08	2.04427e+11	1.80985e+11	21.2132e+08
500	2.5e+08	1.25e+09	2.77115e+11	2.47731e+11	22.3607e+08
550	3.025e+08	1.6637e+09	3.61345e+11	3.29462e+11	23.4521e+08
600	3.6e+08	2.16e+09	4.59549e+11	4.26328e+11	24.4949e+08
650	4.225e+08	2.7462e+09	5.73144e+11	5.38516e+11	25.4951e+08
700	4.9e+08	3.43e+09	7.03543e+11	6.66174e+11	26.4575e+08
750	5.625e+08	4.2187e+09	8.51156e+11	8.10425e+11	27.3861e+08
800	6.4e+08	5.12e+09	1.01715e+12	9.72356e+11	28.2843e+08
850	7.225e+08	6.1412e+09	1.20064e+12	1.15427e+12	29.1548e+08
900	8.1e+08	7.29e+09	1.39807e+12	1.34962e+12	30
950	9.025e+08	8.5737e+09	1.61993e+12	1.55842e+12	30.8221e+08
1000	1e+09	1e+10	1.86636e+12	1.79071e+12	31.6227e+08

Powers of X

X	X^2	X^3	X^.25	X^.5
0	0	0	0	0
50	2500	125000	2.17586e+08	8.30083e+08	7.07107e+08
100	10000	1e+06	1.01948e+09	3.16228e+09	10
150	22500	3.375e+06	4.89391e+09	7.69753e+09	12.2474e+08
200	40000	8e+06	1.60391e+10	1.49336e+10	14.1421e+08
250	62500	1.5625e+07	3.41406e+10	2.80315e+10	15.8114e+08
300	90000	2.7e+07	6.09549e+10	5.19121e+10	17.3205e+08
350	1.225e+08	4.2875e+08	9.82718e+10	8.23173e+10	18.7083e+08
400	1.6e+08	6.4e+08	1.46484e+11	1.2574e+11	20
450	2.025e+08	9.1125e+08	2.04427e+11	1.80985e+11	21.2132e+08
500	2.5e+08	1.25e+09	2.77115e+11	2.47731e+11	22.3607e+08
550	3.025e+08	1.6637e+09	3.61345e+11	3.29462e+11	23.4521e+08
600	3.6e+08	2.16e+09	4.59549e+11	4.26328e+11	24.4949e+08
650	4.225e+08	2.7462e+09	5.73144e+11	5.38516e+11	25.4951e+08
700	4.9e+08	3.43e+09	7.03543e+11	6.66174e+11	26.4575e+08
750	5.625e+08	4.2187e+09	8.51156e+11	8.10425e+11	27.3861e+08
800	6.4e+08	5.12e+09	1.01715e+12	9.72356e+11	28.2843e+08
850	7.225e+08	6.1412e+09	1.20064e+12	1.15427e+12	29.1548e+08
900	8.1e+08	7.29e+09	1.39807e+12	1.34962e+12	30
950	9.025e+08	8.5737e+09	1.61993e+12	1.55842e+12	30.8221e+08
1000	1e+09	1e+10	1.86636e+12	1.79071e+12	31.6227e+08

CmC

CONNECTICUT microCOMPUTER

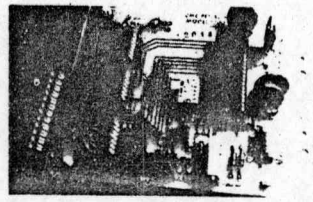
150 POCONO RD. - BROOKFIELD, CT06804 - (203)775-9659

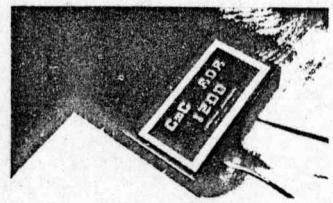
CmC

CONNECTICUT microCOMPUTER

150 POCONO ROAD - BROOKFIELD, CONNECTICUT 06804

(203) 775-9659





RS-232 PRINTER ADAPTER FOR THE COMMODORE PET

The CONNECTICUT microCOMPUTER ADApter model 1200 is the first in a line of peripheral adapters for the COMMODORE PET. The CmC ADA 1200 drives an RS-232 printer from the PET IEEE-488 bus. The CmC ADA 1200 allows the PET owner to obtain hard copy program listings, and to type letters, manuscripts, mailing labels, tables of date, pictures, invoices, graphs, checks, needlepoint patterns, etc., using a standard RS-232 printer.

The CmC ADA model 1200B comes assembled and tested, without power supplies, case, or RS-232 connector for $94.50. The CmC ADA 1200C comes complete for $169.00. Specify baud rate when ordering. (300 baud is supplied unless otherwise requested. Instructions for changing the baud rate are included.)

WORD PROCESSOR FOR THE COMMODORE PET

CONNECTICUT microCOMPUTER now has a word processor program for the COMMODORE PET. This program permits composing and printing letters, flyers, advertisements, manuscripts, articles, etc., using the COMMODORE PET and an RS-232 printer.

Script directives include line length, left margin, centering, and skip. Edit commands allow the user to insert lines, delete lines, move lines, change strings, save onto cassette, load from cassette, move up, move down, print and type.

The CmC Word Processor Program addresses an RS-232 printer through a CmC printer adapter.

The CmC Word Processor Program is available for $29.50.

RS-232 TO CURRENT LOOP/TTL ADAPTER

The CmC ADApter model 400 has two circuits. The first converts an RS-232 signal to a 20 ma current loop signal, and the second converts a 20 ma current loop signal to an RS-232 signal. With this device a computer's teletype port can be used to drive an RS-232 terminal, or vice versa, without modification of the port. The CmC ADA 400 can also be paralelled to drive a teletype or RS-232 printer while still using the computer's regular terminal. The CmC ADA 400 can easily be modified to become an RS-232 to TTL and TTL to RS-232 ADApter. The CmC ADA 400 does not alter the baud rate and uses standard power supplies. The current loop is isolated from the RS-232 signal by optoisolators.

The CmC ADA 400 is the perfect partner for KIM if you want to use an RS-232 terminal instead of a current loop teletype.

The CmC ADA 400S comes with drilled, plated through solder pads and sells for $24.50. The CmC ADA 400B comes with barrier strips and screw terminals and sells for $29.50.

This announcement was composed on a COMMODORE PET and printed on a GE TermiNet using a CmC ADA 1200C printer adapter and the CmC Word Processor Program.

Qty	Description	baud rate	price	total
	CmC ADA 1200B (basic)		$ 99.50	
	CmC ADA 1200C (completed)		$169.00	
	CmC Word Processor Program (cassette)		$ 29.50	
	CmC ADA 400S (solder pads)		$ 24.50	300
	CmC ADA 400B (barrier strips)		$ 29.50	1200
		Subtotal		
	Connecticut residents add 7% sales tax			
	Handling and shipping - add per order		$3.00	
	Foreign air mail - add $5.00 per order			
	Total included with order			

CmC

CONNECTICUT microCOMPUTER
150 POCONO ROAD, ROOM 18
BROOKFIELD, CONNECTICUT 06804

Mail with remittance or charge information to:

NAME

ADDRESS

CITY

STATE ZIP

CHARGE TO: MASTER CHARGE A/C INTERBANK NUMBER Expiration date

Credit card number

SIGNATURE